INTERWOVEN

DataDeploy™ Release Notes

Release 5.5.1

Copyright 1999-2002 Interwoven, Inc. All rights reserved.

No part of this publication (hardcopy or electronic form) may be reproduced or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written consent of Interwoven. Information in this manual is furnished under license by Interwoven, Inc. and may only be used in accordance with the terms of the license agreement. If this software or documentation directs you to copy materials, you must first have permission from the copyright owner of the materials to avoid violating the law, which could result in damages or other remedies.

Interwoven, TeamSite, OpenDeploy and the logo are registered trademarks of Interwoven, Inc., which may be registered in certain jurisdictions. SmartContext, DataDeploy, Content Express, OpenChannel, OpenSyndicate, MetaTagger, TeamCatalog, TeamXpress, TeamXML, the tagline and service mark are trademarks of Interwoven, Inc., which may be registered in certain jurisdictions. All other trademarks are owned by their respective owners.

INTERWOVEN

Interwoven, Inc.

803 11th Ave.

Sunnyvale, CA 94089

http://www.interwoven.com

Printed in the United States of America

Release 5.5.1

Part # 40-00-20-25-00-551-800

Table of Contents

Introduction

This document contains release notes about DataDeploy™ software and is organized as follows:

- **What's new in this release?** – Provides an overview of selected new DataDeploy features.
- **System Requirements** – Provides a summary of hardware and software requirements.
- **Supported Configurations** – Provides a summary supported databases, locales, browsers, and compatibility with other Interwoven® products.
- **Known Issues** – Provides a list of issues that are known to exist in this release.
- **Bug Fixes** – Provides a list of major known issues that have been fixed.
- **Supplemental Information** – Provides late-breaking configuration and usage information about DataDeploy features.
- **Documentation Errata** – Provides a list corrections to previous manuals.
- **Additional Resources** – Provides references to additional information about DataDeploy.

What's new in this release?

DataDeploy 5.5.1 introduces the following key new features:

- Event filtering and persistence—TeamSite® includes a new messaging component. The Event Server is a publish-and-subscribe messaging subsystem that enables you to store events even when a subscriber application is inactive, and to filter events so that subscribers receive only the ones they need. The ability to filter event triggers greatly improves the scalability and reliability of Database Auto-Synchronization (DAS).

  See the *DataDeploy Administration Guide* for details about the Event Server.
- Support for multistore environment—You can deploy data from multiple TeamSite backing stores to different databases. Note the following:
  - You can have up to one instance of DAS running per TeamSite installation.
  - It is recommended that you configure different databases, tablespaces, or database server instances for each TeamSite backing store from which you want to deploy extended attributes and data content records (DCRs).
  - You must modify `ddcfg.template` if you are using wide-table format, `ddcfg_uds.template` if you are using user-defined database schemas, or `ddcfg_uds_custom.template` to deploy custom DCRs to user-defined database schemas. In the file you choose, specify the database settings that are appropriate for your environment before executing the `iwsyncdb.ipl -initial` command.
  - If you want to deploy data from different stores to different databases, you must set up the following file. Note that the word "`store`" here is a variable:

    `dd-home``/conf/drop_`*`store`*`.cfg` so that DAS can process `DestroyWorkarea` and `DestroyBranch` correctly.
  - If you want to deploy TeamSite metadata from different stores to different databases, you must set up the following file. Note that the word "`store`" here is a variable:

    `iw-home``/local/config/mdc_dd_`*`store`*`.cfg` files. To do this, run `iwsyncdb.ipl -mdcddgen` to generate an mdc_dd.cfg file under *`iw-home`*`/local/config` and then rename that generated file to `mdc_dd_`*`store`*`.cfg`

- Support for multibyte characters—DataDeploy is engineered with your global enterprise in mind. This includes internationalizing DataDeploy to support multibyte languages and locales at the operating system, client, and data management levels. See the *DataDeploy Administration Guide* for details about DataDeploy and support for multibyte languages and locales.
- Enhanced deployment times for heavily nested DCRs—DataDeploy 5.5.1 enables you to greatly reduce the time it takes to deploy DCRs that contain many levels of nested elements. For details about how to configure this enhancement, see the *DataDeploy Administration Guide*.

System Requirements

This section describes the hardware and software required to install and run DataDeploy.

Operating Systems and CPU Power

The following table lists the supported operating systems and the recommended CPU type:

Operating System	CPU Power
Windows NT® (x86) 4.0 Server (Service Pack 6a)	600 MHz Pentium® II
Windows® 2000 Server (Service Pack 1)	600 MHz Pentium II
Solaris™ 2.6, 2.7 (Solaris 7), 2.8 (Solaris 8) (32-bit and 64-bit) with Java patches	Ultra5 SPARC™

RAM Requirements

Your server must have at least 256 megabytes (MB) of RAM to operate DataDeploy software.

Storage Requirements

The following table lists the storage requirements for installing and running DataDeploy:

Platform	Environment	Required Disk Space
Windows	with TeamSite	35 MB
Windows	Standalone	100 MB
Solaris	with TeamSite	80 MB
Solaris	Standalone	120 MB

Required Operating System Patches

This section contains information regarding operating system patches. Those patches are provided and maintained by the operating system makers, and they periodically change their Web sites. If you cannot find the operating system patch information you want here, go to the home page of the operating system maker for information related to patches.

Windows

There are no Windows patches required at this time.

Solaris

There are required patches for using DataDeploy with Solaris. Check the Sun web site:

```
www.sun.com
```

for patches required to run Java for your Solaris platform.

Supported Configurations

This section describes supported databases (and their host platforms), browsers, and locales.

Supported Databases

DataDeploy supports the following database/platform combinations.

Database	Windows	Solaris
IBM DB2 (UDB) 7.1	X	X
Microsoft SQL Server 2000 with type 4 JDBC driver	X	
Oracle® 8i	X	X
Sybase® Adaptive Server Enterprise 11.5* U.S. English with case-insensitive sorting	X	X
Sybase Adaptive Server Anywhere® 7.0.1	X	X

* Databases used in deployment must set the "allow nulls by default" option. Refer to your vendor's documentation to set this option.

Release Compatibility

This release of DataDeploy can be used with the following versions of TeamSite, TeamSite Templating, and OpenDeploy®. DataDeploy can be used with TeamSite without OpenDeploy:

TeamSite and TeamSite Templating	OpenDeploy
5.5.1	5.5.1

Upgrading to TeamSite 5.5.1 and DataDeploy 5.5.1

Rerun the `iwsyncdb.ipl -initial` command. It is recommended that you manually drop all existing tables created by DAS, including `iwtracker` and `iwov_idmaps` tables.

Upgrading TeamSite and TeamSite Templating Where DataDeploy 5.2 Already Exists

If TeamSite and TeamSite Templating installations are upgraded on a system that hosts an existing DataDeploy installation with the DataDeploy administration GUI activated, DataDeploy must be reinstalled after the TeamSite and TeamSite Templating upgrades have been completed. That is because modifications made during DataDeploy installation to several files shared by those three applications are overwritten when TeamSite and TeamSite Templating are upgraded.

Supported Browsers

The following browsers are supported for accessing the DataDeploy administration graphical user interface (GUI):

- Internet Explorer (IE) 5.0

Localization

DataDeploy is supported on the following localized operating systems:

Language	Windows NT	Windows 2000	Solaris 2.7 (Solaris 7)
U.S. English	x	x	x
Japanese	x	x	x
French		x	x
German		x	x

Support for Sybase Adaptive Server Enterprise and Adaptive Server Anywhere

Sybase Adaptive Server Enterprise 11.5

DataDeploy supports Adaptive Server Enterprise 11.5 only when deploying to an U.S. English database with case-insensitive sort ordering.

Sybase Server Anywhere 7.0.1

DataDeploy can connect to Sybase Adaptive Server Anywhere 7.0.1 using either the JDBC-ODBC bridge or the native jConnect JDBC driver.

If you are using Sybase SQL Anywhere 5.5 for deployment and would like to start using version 7.0.1, refer to your Sybase documentation for information on how to move existing data from Sybase SQL Anywhere 5.5 to Sybase Adaptive Server Anywhere 7.0.1.

Because DataDeploy relies on information retrieved from database metadata tables, you must install the jConnect metadata accessor classes in the database server that you will use for deployment unless:

- You are deploying to a Sybase Adaptive Server Anywhere database that was installed with TeamSite.
- You are deploying to a Sybase Adaptive Server Anywhere database, created through Sybase Central, with metadata accessor classes installed.

The SQL script `jcatalog.sql` that installs these system objects is shipped with Sybase Adaptive Server Anywhere 7.0.1 and is located in the following directory:

`install_directory``/scripts`

where *`install_directory`* is the root directory in which Sybase Adaptive Server Anywhere 7.0.1 is installed.

Note: If you are creating a new database that you will use with DataDeploy, and if you installed Sybase ASA as part of TeamSite 5.0, `jcatalog.sql` will be in the following path:

Windows: *`iw-home`*`/tools/db/sqlanywhere-7.0.1/scripts`

Solaris: iw-home`/tools/db/sqlanywhere-7.0.1/SYBSsa7/scripts`

Before using DataDeploy, run the `jcatalog.sql` script from Sybase ISQL. This is a one-time operation. Refer to the *Adaptive Server Anywhere User's Guide,* Chapter 19: "Data Access Using JDBC" (in Part 4: "Adding Logic to the Database") for information on how to install the jConnect system objects. Refer specifically to the section "Installing jConnect system objects into a database."

If you are using Sybase Adaptive Server Anywhere 7.0.1 on a Windows system, and if you desire JDBC-ODBC connectivity, set the `db` attribute in the DataDeploy configuration file to an ODBC Data Source Name.

To connect to a Sybase Adaptive Server Anywhere 7.0.1 database using jConnect JDBC driver, set it to:

`hostname``:`*`port`*

where *`hostname`* is the machine on which Sybase Adaptive Server Anywhere 7.0.1 is running and *`port`* is the TCP/IP port number to which the Sybase Adaptive Server Anywhere 7.0.1 database server is listening.

Support for i-net UNA™ 4.02 JDBC Driver When Deploying to a Microsoft SQL Server Database

You can use user-defined database schemas (a new feature in DataDeploy 5.2) to deploy TeamSite Templating data content records to a Microsoft SQL Server database using an i-net UNA 4.02 JDBC driver, which is a third-party type 4 JDBC driver. Deployment using the JDBC-ODBC bridge may not work properly.

Note: If DataDeploy is configured to deploy into a Microsoft SQL Server Database, using the JDBC-ODBC bridge, the SQL Server ODBC version should be 3.70.08.20 or later.

Because Microsoft does not offer a JDBC driver, DataDeploy has been using the JDBC-ODBC bridge as the connection protocol in order to deploy to a Microsoft SQL Server database. The JDBC-ODBC bridge comes with a JDK, provided by Sun Microsystems. Unfortunately, this driver has significant limitations.

Starting with release 5.0, DataDeploy supports the i-net UNA Version 4.02 JDBC driver when deploying to a Microsoft SQL Server database. DataDeploy continues to support connections to SQL Server using the JDBC-ODBC bridge.

However, if you use user-defined database schemas while deploying to a SQL Server database, you can minimize problems by using the i-net UNA JDBC driver rather than the JDBC-ODBC bridge.

The advantages of the i-net UNA JDBC driver include:

- If you running TeamSite on a non-Windows platform, you can deploy data content records or TeamSite metadata (extended attributes) to a SQL Server database running on Windows, without needing to install an ODBC Driver Manager on the non-Windows platform.
- You may use native features offered by SQL Server and not supported by the JDBC-ODBC bridge (for example, the `TEXT` data type).
- You can increase performance, because i-net UNA is a type 4 (pure Java) JDBC driver that directly communicates to the database server whereas the JDBC-ODBC bridge does not communicate directly.

The i-net UNA JDBC driver is bundled with the DataDeploy installation package. To use the driver in DataDeploy:

1. Configure your Microsoft SQL Server to accept connections over TCP/IP using the SQL Server Client Network Utility. Refer to your Microsoft SQL Server Administration manuals on how to configure the database server for accepting connections using TCP/IP protocol.
2. In DataDeploy configuration files and DataDeploy template files (*`dd-home`*`/conf/ddcfg.template`, *`iw-home`*`/config/mdc_ddcfg.template`):
   - Modify the `db` attribute value to "*`hostname`*`:1433?database=`*`dbname`*", where *`hostname`* is either the TCP/IP name or the IP address of the system where Microsoft SQL Server is running, and where *`dbname`* is the name of the database used for deployment.
   - Set the `vendor` attribute value to "`microsoft-inetuna`".

`Example:`

```
 <database db            = "localhost:1433?database=datadeploy"
                          user          = "sa"
                          password      = ""
                         ....
                         ....
                          vendor        = "microsoft-inetuna"
                         ....>
```

3. Repeat the previous step for each `<database>` element in the DataDeploy configuration and DataDeploy template file.

Known Issues

#26011: DataDeploy fails to start when it is installed on Windows 2000 Advanced Server under a directory that has a name that contains a space.

A workaround for this problem is described in "Supplemental Information" on page 14.

#24900: (Solaris only) If the location of iwmount is not "/iwmnt", then DDUI is not able to fetch files from vpath.

A "File Not Found" error message is displayed when an end user attempts to load a DCT in the Map Database to DCT section of the DataDeploy administration GUI on Solaris when the location of the TeamSite mount point is not `/iwmnt`.

Workaround:

1. Specify the location of the TeamSite mount point. To do that, add the following line in the `[locations]` section of `/etc/iw.cfg`:

   `iwmount=`*`location-of-mount-point`*

2. Run the following command:

   `iw-home``/bin/iwreset -ui`

#19286: iwsyncdb.ipl -ddgen fails to validate missing "value-from-element" from dbschema.cfg file when custom defined DCR's are used.

#13630: iwsyncdb.ipl -rmbr might not remove tables from a specified branch.

Although using this command will generate an `***iwsyncdb.ipl DONE***` message in the `iwsyncdb_rm.log` file, this command may not remove tables from a specified branch. If this occurs, remove the tables from your database manually.

#10524: A tablespace attribute value will be used in CREATE TABLE statements only if that statement is generated by DataDeploy.

Appendix B of the *DataDeploy Administration Guide* states that you can add a `tablespace =` option to the `<database>` element in a DataDeploy configuration file. However, if an overriding `CREATE TABLE` statement exists in the DataDeploy configuration file, specifying `tablespace =` in the `<database>` has no effect.

The default `ddcfg.template` and `mdc_ddcfg.template` files both have overriding `CREATE TABLE` statements, which means that adding the `tablespace =` option to the `<database>` element in those files will have no effect.

Instead, edit the `ddcfg.template` and `mdc_ddcfg.template` files so that each overriding `CREATE TABLE` statement has `IN` *`tablespacename`* appended to it.

For example, by default one of the statements appears as follows:

```
CREATE TABLE \$mybasetable (Path VARCHAR(255) NOT NULL,
                        iw_state VARCHAR(30),
   <!-- ### the following is automatically generated by ddconv.ipl ### -->
   $tabledef
   <!-- ### the preceding is automatically generated by ddconv.ipl ### -->
                        CONSTRAINT \$mybasetable^_KEY PRIMARY KEY (Path)
               )
```

You should edit this file so that it appears as follows:

```
   CREATE TABLE \$mybasetable (Path VARCHAR(255) NOT NULL,
                        iw_state VARCHAR(30),
   <!-- ### following automatically generated by ddconv.ipl ### -->
   $tabledef
   <!-- ### preceding automatically generated by ddconv.ipl ### -->
                        CONSTRAINT \$mybasetable^_KEY PRIMARY KEY (Path)
               ) IN mytablespace
```

Bug Fixes

The following issues were fixed as part of the DataDeploy 5.2 Service Pack 1 release and are included in DataDeploy 5.5.1:

All Platforms

#24516: When running DataDeploy on a host where no TeamSite is installed an error message is displayed.

#24476: *_dd.cfg file has incorrect column definition.

#24440: DDUI: standalone config file sets wrong vendor.

#24080: While inserting varying level replicants, DataDeploy skips inserting rows if at least one value is missing in the DCR for a non-nullable column in the table. This doesn't happen when inserting same level replicants. Behavior should be consistent in both cases.

#23868: It takes nearly 45 seconds to deploy one DCR that's heavily nested (up to 6 levels).

For more information on using this fix, see note 11 about the `<database>` section of the sample configuration file in the Configuration File Details chapter of the *DataDeploy Administration Guide*.

#23760: DataDeploy doesn't handle multiple TEXT nodes for an element correctly while deploying custom DCRs.

#23738: When deploying two DCRs of the same type in one single deployment, incorrect inserts are being performed.

#23715: CREATE TABLE statement constructed by DataDeploy for the temporary table to perform sorting doesn't honor "allows-null" attribute set for columns when the back end is DB2.

#23703: Inconsistent behavior when replicant order column is specified.

#23686: Request for more error/exception handling for external tuple preprocessor interface and external data source interface handling.

#23501: NullPointerException occurs while deploying a custom DCR.

#23305: When mapping to an existing schema, any unmapped columns get a null value-from-field.

#23236: Deployment to a TIME data type column fails.

#23201: When DataDeploy deploys file content to a CLOB column using the is-url attribute value, the deployed content is of incorrect size.

#23017: DataDeploy doesn't report certain transient exceptions that occur while trying to parse DCRs.

#22624: dbschemagen doesn't generate the value-from-field's correctly for container tags.

#22583: Standalone deployment fails under special circumstances. If the dbschema contains only one group in which all columns are replicants and if you are deploying custom DCRs, exceptions occur.

#22417: When the root group has only one column and if standalone deployment is performed with delete-tracker set, updates fail.

#22044: When a primary key is not supplied in a dbschema configuration file non-root group, DataDeploy fails and hangs.

#21382: DataDeploy-OpenDeploy sync:Frequent "com.interwoven.sj100.sciface.sci.CloseSession(Native Method)" errors when running ddsync.ipl.

#14233: The DataDeploy 5.0 installer should install dd-home/iw-perl only if iw-home/iw-perl does not exist.

#12843: iwsyncdb.ipl -initial <vpath> enumerates sibling workareas too.

#10384: Registry entries are not being removed when uninstalling OpenDeploy and DataDeploy.

Solaris

#22118: DataDeploy GUI requires write permission for the iwui user on the workarea for which DAS needs to be set up using the DataDeploy GUI.

Supplemental Information

Installing DataDeploy on Windows 2000 Advanced Server

The following information applies only when DataDeploy is installed on Windows 2000 Advanced Server.

DataDeploy fails to start if it is installed under directories that contain spaces in their names (for example: `C:\Program Files\Interwoven\DataDeploy`). If you need to install DataDeploy in a directory that contains a space, you must do the following. Note that you must have administrator privileges to perform these tasks:

1. Open the Registry Editor

   Select **Start** > **Run**, and enter **`regedt32.exe`**.

2. Go to `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services`.

3. Select `iwdatadeploy` in the left pane.

4. Double click `ImagePath` in the right pane.

   The existing value is displayed in an edit box.

5. Enclose the existing value in double quotes. For example, if the existing value is:

   `C:\Program Files\Interwoven\DataDeploy`

   change it to:

   `"C:\Program Files\Interwoven\DataDeploy"`

6. Start the DataDeploy service.

This problem does not occur when DataDeploy is installed under directories without spaces.

Synchronized Deployment Using DataDeploy and OpenDeploy

When an incorrect encoding is specified in the customization section of the `ddsync.ipl` script, the script actually will abort. However, OpenDeploy is not informed of this from the Deploy and Run process and OpenDeploy will proceed as if the script execution succeeded.

The `ddsync.ipl` needs to validate the encoding and inform OpenDeploy of an error if the encoding is invalid. To enable this, you must apply the following fix to the `ddsync.ipl` script:

Insert the following perl code snippet after line 300 as follows:

```
     if ($input_encoding ne "NONE")
     {
         ...
         require TeamSite::I18N_utils;
# begin snippet
         my $encoding =TeamSite::I18N_utils::promote_encoding($input_encoding);
         if ($encoding ne 'UTF-8' && $encoding ne 'UTF-16') {
             my $map  = Unicode::Map->new( $encoding );
             if (! $map ) {
                 print OUTFILE "ERROR: Unrecognized encoding specifed in script\n";
                 my $tell_od_of_err = qq[<response code="-2" />\n];
                 print STDOUT $tell_od_of_err;
                 close OUTFILE;
                 exit 1;
             }
         }
# end snippet
         ...
     }
```

Database-to-database Deployment

DataDeploy supports database-to-database deployment only when data from one table is deployed to one destination table. You cannot use user-defined schemas for database-to-database deployments. For such deployments, use the database replication tools supplied by your database vendor.

Internationalization

All configuration files generated by DataDeploy are UTF-8 encoded. Ensure that you save those files as UTF-8 if you edit them. See the *DataDeploy Administration Guide* for more details about internationalization. On Windows systems, you can use Notepad or Wordpad to edit those files because those text editors support opening and saving files that are UTF-8 encoded.

DataDeploy and DAS

Support of File-System Events

The Database Auto-Synchronization module (DAS) automatically updates the appropriate base and delta tables whenever a file containing extended attributes is moved or renamed from the command line or the file system interface. This is in addition to the table updates associated with file changes initiated from the TeamSite GUI. This feature is fully configured after you enable DAS. For more information, see the DAS section of the *DataDeploy Administration Guide*.

DAS Out-of-Sync Conditions

The following scenarios are not supported by DAS and will cause out-of-sync conditions with the database:

Scenario 1: Using the `iwextattr` command-line tool to add or delete extended attributes on a file if DAS is set up for extended attributes with wide tables.

Scenario 2: Manipulating a data content record (for example, renaming, editing, moving, deleting, and so forth) from the command line or file system interface.

Scenario 3: Shutting down the DataDeploy daemon and then performing any kind of extended attribute or file manipulation. In this situation, you must either restart the DataDeploy daemon or perform a manual resynchronizing deployment (for example, by running `iwsyncdb.ipl -resyncwa` or `iwsyncdb.ipl -resyncbr`).

Omitting State Columns

You may omit the State column from any table. It is unlikely that you would need to omit it from a delta table. Some scenarios could necessitate omitting it from a base table.

Updates Performed as a Single Transaction

When a base table is updated through the submit scenario described in "Updating a Base Table" in the *DataDeploy Administration Guide*, updates to all affected tables are done as a single transaction.

DAS and System Performance

- Filtering unwanted table updates

  After DAS is configured, it updates table entries whenever a file containing extended attributes is moved or renamed. However, in some situations, it might not be necessary or optimal to update all tables. For example, if an application automatically generates a backup file whenever the main version of the file is edited, it might not be practical for the backup file changes to result in table updates. You can use DataDeploy filters as described in the *DataDeploy Administration Guide* to control which changes will result in table updates. For example:

  ```
  renamefse_filter="^[^~]*$"
  ```

  allows all rename events except for events with a "~" character. Microsoft Word renames files to and from temp files with a "~" prefix.
- If you use DAS with file-level events instead of manual deployment for moving or renaming files, performance may degrade up to 15 percent.

Additional Resources

Readme Files

`README` files, located in many of the installed directories, provide additional examples or information on available features.

Information Available on the Web

These release notes and any other available late-breaking information are located at:

`http://support.interwoven.com`

Information on the CD-ROM

Release notes and the following manuals are available as PDF files in the `doc` directory on the CD-ROMs shown below:

DataDeploy CD

- *DataDeploy Administration Guide*
- *DataDeploy User's Guide*
- *DataDeploy Release Notes*

TeamSite CD

- *TeamSite User's Guide*
- *TeamSite Author's Guide*
- *TeamSite Administration Guide*
- *TeamSite Command-Line Tools*
- *TeamSite Backing Store Conversion Guide*
- *TeamSite Workflow Developer's Guide*
- *TeamSite Release Notes*
- *TeamSite Templating Developer's Guide*
- *TeamSite Templating User's Guide*
- *TeamSite Templating Release Notes*
- *TeamSite Templating: VisualFormat Developer's Guide*